Recurrent Perceptron

Trevor Davis Ducharme

October 9, 2024

1 Introduction

This document provides a mathematical overview of the basic perceptron model.

2 Perceptron Model

A recurrent perceptron is a simple classifier that takes into account the previous classifications. The output of the perceptron is given by:

$$r = [x_1, x_2, \ldots, x_n, p]$$

$$z = \left(\sum_{i=1}^{n+1} w_i \cdot r_i\right) + b$$

$$y = A(z)$$

where:

- A is the activation function
- $\mathbf{w}$ is the weight vector
- $\mathbf{x}$ is the input vector
- b is the bias term
- r is the input vector with the previous activation appended
- p is the previous activation

3 Training the Perceptron

The perceptron is trained using the following equations:

3.1 Necessary Derivatives

$$\frac{\partial y}{\partial z} = A'(z)$$

$$\frac{\partial z}{\partial \mathbf{w_i}} = \mathbf{r_i}$$

$$\frac{\partial z}{\partial b} = 1$$

3.2 Updating the Weights

The weights are updated as follows:

$$\mathbf{w} \leftarrow \mathbf{w} + \Delta\mathbf{w}$$

where:

$$\Delta\mathbf{w_i} = -\eta \cdot L'(y, \hat{y}) \cdot \frac{\partial y}{\partial z} \cdot \frac{\partial z}{\partial \mathbf{w_i}}$$

Here, η is the learning rate, L is the loss function, and $\hat{y}$ is the target output.

3.3 Updating the Bias

The bias term is updated as follows:

$$\mathbf{b} \leftarrow \mathbf{b} + \Delta\mathbf{b}$$

where:

$$\Delta\mathbf{b} = -\eta \cdot L'(y, \hat{y}) \cdot \frac{\partial y}{\partial z} \cdot \frac{\partial z}{\partial b}$$